ASW DETECTION PROBLEM: A TACTICAL WAR GAME ON THE CDC 1604 COMPUTER

ROBERT H. YERBURY
and
ARTHUR H. CUMMINGS, JR.

ASW DETECTION PROBLEM

A TACTICAL WAR GAME ON THE CDC 1604 COMPUTER

* * * * *

Robert H. Yerbury

and

Arthur H. Cummings, Jr.

ASW DETECTION PROBLEM

A TACTICAL WAR GAME ON THE CDC 1604 COMPUTER

by

Robert H. Yerbury

Lieutenant Commander, United States Navy

and

Arthur H. Cummings, Jr.

Lieutenant Commander, United States Navy

Submitted in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE

United States Naval Postgraduate School
Monterey, California

1 9 6 1

ASW DETECTION PROBLEM

A TACTICAL WAR GAME ON THE CDC 1604 COMPUTER

by

Robert H. Yerbury

and

Arthur H. Cummings, Jr.

This work is accepted as fulfilling

the thesis requirements for the degree of

MASTER OF SCIENCE

from the

United States Naval Postgraduate School

ABSTRACT

In this highly technical era, it is becoming increasingly expensive to prove or disprove a method of operation by means of employing actual forces. In addition, data extraction from the reports of the actual exercises is difficult and always time-consuming in its analysis. There is a need for a simple and flexible method for the evaluation of gross parameters and the feasibility of tactical dispositions. Industry is making increasing application of the general purpose digital computer to aid management in its decision making processes. Some progress has been made in the adaption of the computerized techniques to the field of war gaming.(1) This limited application has invoked considerable interest and has demonstrated that computerized war games are feasible. This thesis proposed such a method and as an example of the detail of the method and its uses, has been applied to an ASW Sortie Exercise. The salient features of this exercise are time, data available for extraction and the ease of manipulation of the parameters. In addition to the investigative procedures for gross parameters, emphasis must be placed on the gaming aspects for training and planning.

Note: Superscribed numbers in brackets refer to numerical references in bibliography. Numbers without brackets indicate footnotes.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

TABLE OF SYMBOLS

R - Range in grid units (octal) from sensor (destroyer, helicopter, etc.) to submarine

R' - Gross estimate of range $(\Delta x + \Delta y)$

R^* - Maximum range at which evasion or detection can possibly occur

Δx - Octal grid difference in "X" direction between submarine and sensor positions

$\left.\begin{matrix}\Delta \dot{x} \\ \Delta \dot{y}\end{matrix}\right\}$ - Incremental movement per unit time of submarine or sensor

C_D - Destroyer present course

C_i - Submarine present course

C_z - Submarine course from present position to score area

$\left.\begin{matrix}X_i \\ Y_i\end{matrix}\right\}$ - Grid coordinate of a unit where i refers to type of unit; D = destroyer, S = submarine, H = helicopter

V_s - Submarine present speed

r_i - Random number (pseudo)

P - An octal number representing probability from interval (0, 1)

σ_D - Destroyer orientation (code number of search pattern)

σ_H - Helicopter orientation (code number of search pattern)

t - Game time

t_c - Cycle time

Y_p - Detection anomaly parameter (If $Y < Y_p$ detection probability is reduced to simulate thermal layer)

I

INTRODUCTION

1.1 PURPOSE:

The purpose of this thesis is to demonstrate a method for using a general purpose digital computer and gaming methods to play a War Game which incorporates the features of both "Computational" and "Maneuver Board" type war games.(2) It is hoped that the single game presented to demonstrate the method will be found to incorporate the uses and advantages of the two types mentioned, and further, to eliminate some of the disadvantages of each. That is, we plan to show how a computer can be used to speed up a "Maneuver Board" game, and how a maneuver board can be employed to strengthen, and to verify if desired, a "Computational" game.

1.2 GENERAL DESCRIPTION:

In order that the computer method may be demonstrated most effectively, a game of some complexity, yet of limited scope was desired. Antisubmarine warfare conditions encountered by naval forces upon exit from a sheltered harbor were found to be a most satisfactory subject. The selection of this particular problem does not imply limitation of the method presented herein, but rather reflects the limitations of time and facilities imposed upon the authors.

Basically the game is envisioned as of the two person zero-sum variety, with umpire and the classic Red and Blue protagonists. However, the game theory aspects may be suppressed and the play made for investigation of gross physical parameters by the Monte Carlo Method. The game is considered to be useful in three separate forms of application, which we shall identify as Playing Modes:

(a) Analysis Mode - A statistical analysis may be made of the variation in detection capability of a given force for several values of selected parameters, such as the sonar detection range curve.

(b) Evaluation Mode - A statistical comparison may be made of the effectiveness of alternate ASW barriers, or, the weaknesses of a given plan may be investigated by multiple and diverse submarine penetrations.

(c) Training Mode - The game theory capabilities may be exploited by having Red and Blue determine strategies independently and secretly prior to play of the game. Red and Blue might be staff planners or school trainees. Here the game may be played once with a detailed analysis of the game history, or played several times to determine mean or average outcomes.

1.3 THE MANEUVER BOARD

Appendix A illustrates the layout of the maneuver board used to play this game. It may be simply constructed to duplicate roughly any land-water (or water only) area desired. The score area may be formed in any desired rectangular shape. The distance and time scales[1] may be set to any values desired. The configuration selected was chosen so as to illustrate all of the features of the game, and is not intended to represent any geographic area, display any specific tactic or formation, nor to simulate any actual service equipments. The physical size of the maneuver board is arbitrary so long as the time units and dimensions properly represent the scale of the game.

Actually two grids are employed. An "Overlay Grid" (see Appendix B) is required to transform the data from the playing grid of Appendix A,

[1]A discussion of units and time will be presented in section 2.2.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

which may be employed in various forms and orientations (i.e., rectangular as illustrated, or polar), to numerical values compatible with the input and output system of the computer. The Overlay Grid may be in the form of an overlay transparency or superimposed in different color on the playing grid. The grid divisions may be made as fine as desired, but the numerical values must be in OCTAL numbers.

Using the playing grid, Red and Blue position forces and determine search patterns or courses and speeds in response to the playing conditions. The computer operator uses the overlay grid to determine computer input data, enters the data, sets up the program, and if using the automatic method of computation, plays the game and announces the results. The methods and particulars of this procedure constitute the body of the thesis.

1.4 GAME RULES:

The number of submarines which are detected or fail to reach the score area within the designated game time is considered Blue's score. The number reaching the score area constitutes Red's score.

Given the exploitation of the full potential of this game, there are three possible computer modes:

(a) Automatic - in which after initial assignment of Red and Blue positions, assured sonar range, sonar operation factor and water temperature gradients, the computer controls all moves, makes the evasion decisions previously selected by the Red Commander, serves as umpire where chance moves are involved, and tallies all scores.

(b) Semi-automatic - which is essentially as (a) above except the computer controls moves until a critical event such as a submarine detection by a destroyer or a destroyer detection by a

submarine. When such an event occurs, the computer halts and allows either Red or Blue to make an appropriate decision before continuing.

(c) Manual - at the end of each unit of game time the computer halts and allows both Red and Blue to make their moves before continuing. In this respect, it would operate only as an umpire and position keeper.

Each of the above modes has its advantages and disadvantages. Mode (a) is fast. In a typical game with 19 sensors and 8 submarines, 90 game minutes required only 15 seconds (computer time) to play. Its disadvantage is the stereotyped decisions which are programmed for Red and Blue. Mode (b) eliminates the disadvantage of mode (a) but has the disadvantages of increased playing time and lack of control over initial moves until a critical event occurs. Mode (c) eliminates the requirement to pre-determine decisions and movement of units disadvantages but greatly increases playing time for a game. Modes (b) and (c) are not investigated in this paper but the program contains indications of how they could be incorporated.

For the initial setup, the umpire will announce all game data, such as the assured sonar range, the position of the temperature gradient, and the game time limit, to both Red and Blue. He will inform Red and Blue individually of the forces at their disposal and the operation factors as needed. Red and Blue will deploy their forces as they desire. Using this initial data, the computer will run the game and inform each player of the final score. In computer storage, there is a wealth of data for extraction. Certain cells inform Red what sensors his submarines detected. Others will inform Blue which sensor detected the

submarine, what time and at what range. For the overall game analysis the entire history of the game can be extracted.

1.5 ASSUMPTIONS:

Certain basic assumptions are built into ASW Sortie game to increase its simplicity with little detraction from the principles by which it was conceived. The ASW Sortie game board is primarily a three dimensional area which must account for depth as well as horizontal distances. Preliminary computations revealed that normal depths around harbor entrances are such that the slant range to the target could be approximated by the horizontal range. Such an assumption permitted the reduction of the game to two dimensions which greatly simplified computations.

Other assumptions included are:

a) The closest threat is always considered paramount for the submarine. Therefore, the submarine will evade only the nearest sensor that it has detected.

b) High speeds limit the acoustical capability of a submarine's detection equipment. Certain conditions may exist where speeds of 20 or 25 knots have been entered into the submarine data. The assumption is made that this does not limit a submarine's detection capability.

c) No delay method is employed in submarine movement change. This does not introduce any appreciable error as the changes are not radical.

d) The sonar operators are constantly on the alert for contacts. In any unit game time interval it is assumed that the operator may detect any target within his circles of probabilities.

d) The barrier is perfectly synchronized in that all sensors

employ the same movement time to compute their new position for the next game minute.

f) The standard figure-eight patrol used by destroyers on barrier stations does not use advance and transfer in the turns. (See Appendix F)

g) A sonar search can be reduced to circles of probability of detection and these circles are not modified by any external effects such as baffles, speed etc.

h) Both detection and classification can be accomplished in the same game minute. This assumes that an operator only requires two or three "pings" to identify a target.

i) The effects of adverse weather are not considered.

j) Fixed wing aircraft are not employed in this game.

k) Submarines are limited to 12 courses commencing at grid North and speeds from zero to 25 knots in increments of 5 knots.

l) Submarine aspect does not affect detection.

m) The state of training of all units is equal.

n) Detection is based exclusively on range. No acoustic anomalies are considered in the computation of the probability of detection or classification.

II

COMPUTATION METHODS

2.1 GRID:

As mentioned previously, the primary considerations in the design of the game are speed and flexibility. To take full advantage of the characteristics of a digital computer, it was decided to avoid iterative type procedures for computation and to employ, whenever practicable, table look-up methods. This decision dictated the use of discrete values for coordinates, ranges, speeds and time units.

The overlay grid is shown in fine detail in Appendix C. This hexagonal grid has been used previously in war games of a different character.[1] It is particularly well suited for the purposes of this game. First, the individual hexagonal cell is a close approximation to a circle of allowable position error from the center, if the size of the hexagonal cell is appropriately chosen. Second, the hexagonal grid permits radial measurements in cell units which are virtually independent of direction. Such cell measure can be used for incremental coordinate changes and for range computation. We shall hereinafter refer to distance measure in cells. The advantage of this nomenclature is more apparent when we consider that the real world value (i.e., scale) for a cell may be arbitrarily assigned.

Although the hexagonal grid is uniquely suited for an oblique coordinate system, some additional complexity in computation was considered acceptable in order that we might employ the familiar rectangular coordinate system and establish a north-south east-west grid axis. Thus it

[1]SWAP, a maneuvering board game for air warfare developed at the RAND Corporation, Santa Monica, California.

will be seen that for a given distance in the X (east-west) direction a zig-zag count of cells (or numerical difference) must be multiplied by cosine 30° in order to obtain the true difference in cell values (see appendix C). We are concerned with this correction only in the computation of range.

By use of discrete cells precomputed tables may be stored in memory for rapid computation of range, movement and detection. To illustrate the method, consider the computation of range from a sensor located at cell 00234,00150 (X,Y coordinates) and a target at cell 00256,00133.

$$\Delta x = X_s - X_k = 256 - 234 = 22$$

$$\Delta y = Y_s - Y_k = 133 - 150 = -17$$

Using the cosine 30° correction the range computation by numerical methods is -

$$R = \left[(.886\,\Delta x)^2 + \Delta y^2\right]^{\frac{1}{2}} = \left[(.886 x 22)^2 + (-17)^2\right]^{\frac{1}{2}}$$

By utilizing the table look-up method this may be reduced to -

1. Multiply Δy by Δy conventionally
2. Enter a table obtaining $(.886\,\Delta x)^2$ for each Δx entry
3. Add the above two values
4. Search a table containing values for the sum versus values of R, rounded off to a discrete number of cells.

The above computation can be accomplished in 100 to 180 microseconds with the CDC 1604 computer. Inasmuch as approximately three hundred range computations may be necessary for one play of the game, the saving in total time for this computation alone over more sophisticated methods is appreciable. Additional time is saved by the simple expedient of adding Δx and Δy prior to range computation and rejecting consideration of situations in which the sum is greater than a given threshold value.

[illegible]

This amounts to a search for critical events.

Sensor and submarine movement are likewise accomplished rapidly by table look-up of incremental grid values. For a given submarine course and speed $\Delta\dot{x}$ and $\Delta\dot{y}$ are obtained from pre-computed tables and applied to the previous coordinates. Sensor movement employs search patterns achieved by applying $\Delta\dot{x}$ and $\Delta\dot{y}$ from designated search plan tables to fixed reference coordinates in accordance with cycle time values. Submarine movement is complicated by the fact that odd numbers of cells per unit time interval produce ambiguities on odd numbered courses of the compass rose of Appendix C (i.e., not perpendicular to the faces of the hexagonal grid cell). This is resolved by alternating the dichotomous choice according to odd or even cell numbers and odd or even total elapsed time. This requires multiple tables but computation time is again minimal, on the order of 250 microseconds per move.

Although fixed-wing aircraft search patterns were not programmed, the same table look-up movement method appears applicable. The speed differential may be handled by treating a fixed-wing aircraft as a line of sensors, that is, the detection circles would be advanced along consecutive cells during each time interval. Although the addition of a major sub-routine would be required, by employing proper methods the computation time should not be greatly increased.

2.2 SCALE:

As indicated in the preceding section the scale of this game is arbitrary. Accordingly a tactical game of slow movement will probably dictate a measure of cell diameter in yards and a unit increment of time as a minute. For fast movement the cell diameter may be chosen to be a mile, or ten miles. For a game covering great distances at high speed, the time unit may be chosen to be an hour (or a fraction of an hour).

The distance and time scales have meaning only to the user, the computer considers all computations sequentially in the same manner regardless of the assigned scales.

For the purposes of the particular ASW Sortie game chosen, 160 yards was selected as the cell diameter value. This closely approximates the length of a standard submarine, and also represents the distance traveled in one minute at five knots. The unit time interval was then obviously chosen to be one minute.

Inasmuch as reference is made to various "times" the following definitions are appropriate:

Game time - Real world time (e.g. minute, hour). A unit of game time is an interval in which a complete sequence of events occur.

Game time limit - The total game time desired for play of the game.

Cycle time - The game time used in search pattern sequencing.

Computation time - The number of microseconds required by the computer to perform a given operation.

Stress time - A variable parameter introduced in simulating the decision stress imposed by imminent termination of the game time limit.

2.3 RANDOM NUMBER GENERATOR:

The procedure for the generation of random numbers employed in this game is a modification of a method described and used by the programmers of the System Development Corporation of Santa Monica, California. What is desired is a random number in the interval (0,1) which could be compared with the probabilities described in section 2.5 to determine detection and classification.

The generator takes an initial input random number and squares it. The product is shifted and the middle portion extracted and stored for use. Safeguards are built in to prevent negative numbers and zero from being stored.

Tests have been conducted on the generated numbers. No replications were noted in over a million generations. If inspection of the random numbers generated in each game is desired, the procedure is given in Appendix E.

The random number generator is a sub-routine which is reached by means of a "return jump". The generator can be replaced by any proven random number routine of the player's choice without altering the main program.

2.4 MOVEMENT:

The movements of sensors require certain simplifications. The speed of the destroyers is considered constant at fifteen knots, or three cells each game minute. For the destroyers, five orientation tables may be entered; the axes of the search patterns are 090°-270°, 030°-210°, 060°-240°, 120°-300° and 150°-330°. To move the destroyers, difference values ($\Delta\dot{x}$ and $\Delta\dot{y}$) according to time sequence are obtained from the orientation table specified for the particular destroyer being moved. These values are added to the reference position to obtain a temporary position. Therefore, destroyers always cycle with respect to the reference positions entered by the Blue Commander. Modification or complete change of destroyer speed and search pattern may be readily accomplished by altering table values or entering new tables (see Appendix F).

The movement of helicopters is accomplished in a manner nearly identical with the destroyer movement. Provision is made to remove the

helicopter from consideration during the enroute segments of the search pattern. For the helicopters, three orientation tables may be entered; a north-south line, an east-west line, and a triangular pattern are available.

The submarine movement requires similar simplifications. The new position is computed from the last position. The movement table output is dependent primarily on course and speed, but is also dependent on grid position and game time. The submarines are limited to 12 courses commencing at grid north and every 30 degrees thereafter (see compass rose Appendix C). Submarine speed range is zero to 25 knots, or zero to 5 cells per game minute. In addition, provisions have been made to prevent the following:

(a) If a submarine is on a course which would take it off the playing board, the computer will enter a computed course to the score area in its present course when the submarine reaches the boundary.

(b) If the submarine's course is such that on the next move it would enter shallow water, the computer will select a course clear of the obstacle before advancing the submarine.

(c) Normally, to accomplish (b) above, the computer enters the course to score first, then if that fails enters grid north for the submarine course. To prevent the submarine from continuing north, as the computer will not change this course unless the submarine is attempting to evade or for (a) and (b) above, the computer will allow the submarine to move on this course for only one game minute before it will enter the course to score for the present course. If it is still not clear of this obstacle, it will continue north for as many minutes as necessary.

(d) It should be noted that the submarine movement occurs after the evasion decision. If shoal water or grid limits require course and speed change, the evasion decision is negated.

2.5 DETECTION AND CLASSIFICATION:

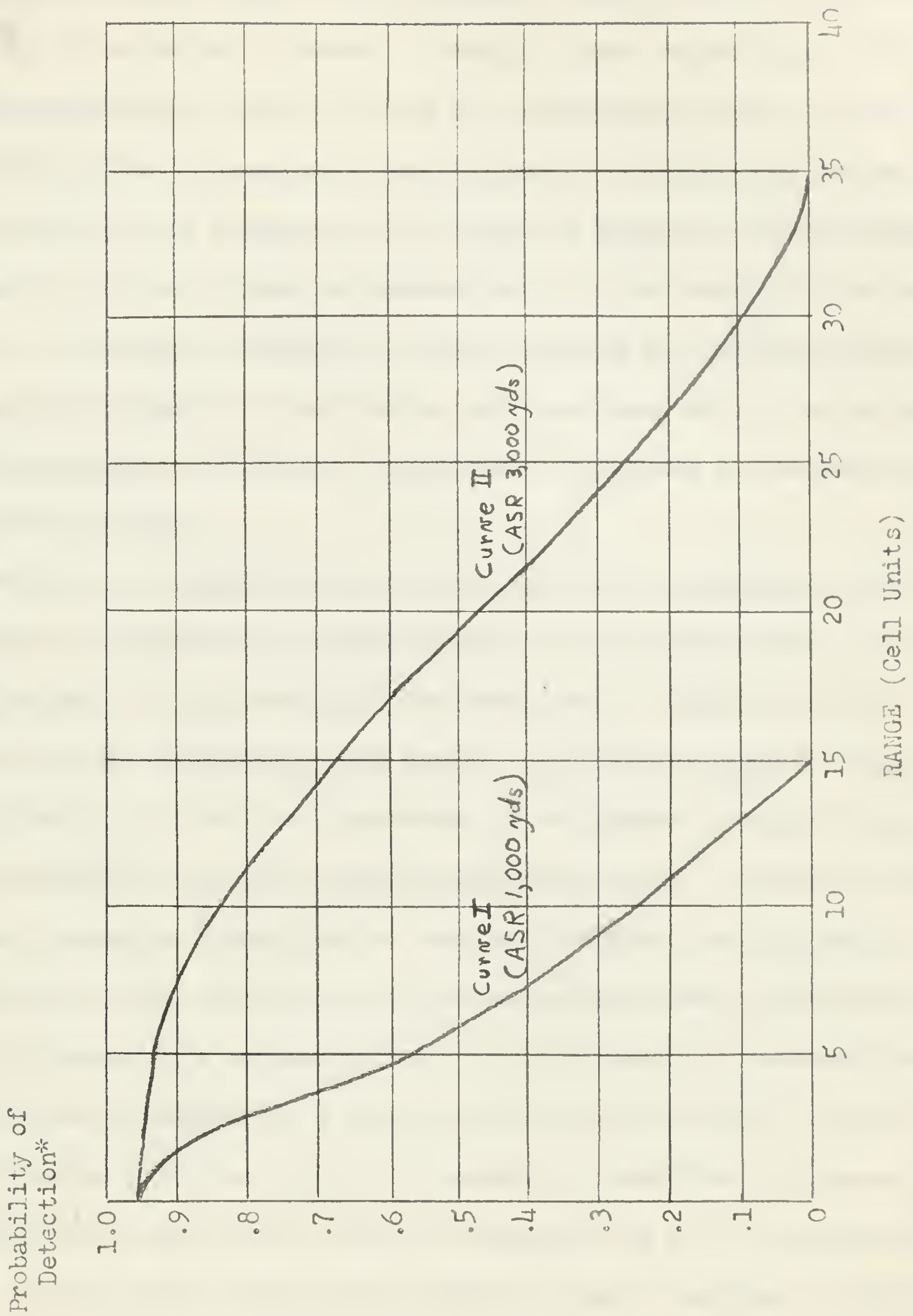
It is appropriate to explain the methods of detection and classification more fully. Desiring an unclassified status for this game, it was decided that pseudo curves or probability versus range would be preferable as this would demonstrate the necessary principles while still maintaining an unclassified status. These pseudo curves, as shown in Figure 2-1, need not be monotonically decreasing.

Two typical detection curves for this game are in Figure 2-1. Range in number of cells is plotted along the abscissa and probability along the ordinate. Equivalent CDC-1604 octal values for the probability are listed alongside. It is this 16 digit value that is entered in the table and used for comparison with the generated random number to determine detection and classification.

In accordance with accepted usage a detection curve is designated by the range at which there is a 50% probability of detection; this is called the assured sonar range.

It is assumed for this game that circles of detection probability are inscribed around the sensor's present position. No attempt has been made to change the probability circle to conform with the typical sonar pattern in which such anomalies as baffle areas distort the circular pattern. In addition, it is assumed that the helicopters are equipped with scanning sonar and not the searchlight type.

To construct a detection table from a detection curve as shown in Figure 2-1, the octal equivalent (see Appendix I) for each discrete



* See Appendix I for conversion to octal table values.

Figure 2-1. Detection Curves

probability is entered in the cell representing the range. The range is the number of cells greater than the initial memory location. For example, if it is desired to enter a detection curve representing 1000 yards assured range (Curve I, Figure 2-1) commencing at cell 11,000, then 3,650,000,000,000 is inserted in cell 11,000. In cell 11,001 the next octal equivalent for a range of 1 cell unit is entered. Each equivalent is entered until octal zero is reached which in this case would be cell 11,015. In this way a detection curve is reduced to a detection table and inserted in memory for extraction and comparison with a random number.

Classification tables are computed and introduced in the same way as the detection tables.

Detection or classification is accomplished by comparison of the probability of detection or classification from the above tables with a random number. If the probability of detection or classification is greater than the generated random number, detection or classification is accomplished. In order for a destroyer or helicopter to tally a score both the detection and the classification must occur. A positive detection and a negative classification does not produce a scoring tally.

Under the same acoustic conditions submarines have a greater capability of detecting a surface craft. For this game it is assumed that the submarine capability is 3 times greater than the sensor. In order not to require additional tables for submarine detection, the sensor detection table is used but to take into account this 3 to 1 advantage, one third of the actual range to the sensor is used to compute the probability of detection of the sensor by the submarine. In addition, the assumption is made that if a sensor detects a submarine, the submarine has automatically detected the sensor. Thus, the submarine will normally

commence his evasion routine beyond possible detection by a sensor. As a further simplification, the submarine will evade only the closest sensor.

2.6 EVASION:

The evasion routine permits the submarine to determine its position relative to the sensor being evaded and in addition permits determination of relative movement according to the type of sensor and range. The submarine has three different evasion subroutines depending upon the range to the nearest sensor.

Long Range:

(a) If the range is greater than 5,000 yards, if the sensor is a helicopter, if the submarine speed is greater than a selected value, if stress time has been reached.

Intermediate Range:

(b) If the range is between 3,200 and 5,000 yards.

Short Range:

(c) If the range is less than 3,200 yards.

Certain evasion decisions are programmed for each of the above range intervals. If the Red Commander does not agree with the evasion decisions, he may change the modules concerning those decisions. Appendix H illustrates part of the intermediate range evasion sub-routine.

III

PROGRAMMING

3.1 METHODS

The war game was programmed entirely in Control Data Corporation Model 1604 Computer, "Assembly Routine" language,[3] which is essentially a phonetic aid to numerical coding. The program is modular to some degree in that major sub-routines may be cut-out of the program and the remainder will operate. However, the separate modules are stored consecutively to conserve space for data storage, and major changes will require jumps to patching locations.

A "backbone" or basic routine which makes the initial entries, moves all submarines and sensors, scores, and stores data, constitutes the major part of the program. The detection and evasion routines are the major sub-routines which complete the game. These in turn are composed of sub-routines.

Since discrete integer values are used throughout the war games, the computational procedure is greatly simplified, and the programming is generally straight forward. In the decision processes required in movement and evasion sections of the program, comparison commands, such as "threshold search", and "jump" (if contents of registers are of certain sign), are used primarily. Cell "packing" has been done conventionally, except that heavy emphasis was placed on address substitute and load-and-store commands to avoid the slower shift commands.

Time keeping is accomplished by index advance after all submarines have been checked against each sensor, then moved. This results in a "critical event" time-keeping method; that is, game time proceeds rapidly through uneventful moves and more slowly during active phases of the game.

3.2 DATA FLOW:

Although the data flow sequence is nearly self-explanatory upon consideration of Appendix D, a brief description of the major events may be helpful. The program operates as follows:

1. Operator enters required data and starts computation.
2. Appropriate tables are selected, addresses and indexing numbers inserted and certain bookkeeping operations are accomplished. Computation starts for the first interval between zero and one minute checking the first submarine against the first sensor.
3. A rough range calculation is made (rapidly) to determine if the first sensor is definitely outside of contact range. If out of range, a jump is made to the decision box "checked all sensors". If within possible action range, the actual range is computed. A random number is then generated and checked against a number representing the sensor's probability of detection at that range. If the random number is larger no detection occurs, and the submarine is subjected to a similar computation to determine if it has detected the sensor. If the sensor does detect the submarine another generated random number is compared to a table representing the "operator factor" of the sensor (this may be regarded as representing the capability of the sensor to classify or otherwise eliminate the submarine from action) versus range. If this random number is greater the submarine survives and is also credited with detecting the sensor as a result of the action. See Appendix J for an illustration of this method. Otherwise the submarine is scratched from the game, a detection is credited to Blue, and the sequence is jumped to the decision box "moved all subs".

[illegible]

[illegible]

[illegible]

4. If detection did not occur in the above sequence the sensor number is checked, and if all have not been checked, the sensor number is advanced and the process in 3 above, is repeated until detection occurs or all sensors have been checked.

5. If undetected the submarine selects the nearest sensor it has detected. Evasive action is selected by a series of decisions which consider the relative location of the sensor and submarine, the position of the submarine with respect to the score area, and if the sensor is a destroyer the projected course is considered. New course and speed are entered as required. See Appendix H for a fragment of the evasion routine illustrating the decision method.

6. Using submarine course and speed, plus decision rules as regards odd or even cell positions and game time to resolve ambiguities, a new submarine position is computed.

7. Before movement the new submarine position must be checked to determine if it is within the playing area grid limits, or outside of the shoal water line. If within the score area, the submarine is deleted from the game and a score credited to Red. If the new position is improper, various sub-routines are used to change course and speed, and another new position is computed and checked.

8. When an acceptable new position has been found the submarine is advanced. After this occurs, a course to score is computed and stored for use during the next move, and certain bookkeeping functions are performed.

9. In the decision box "moved all subs" the submarine number is

advanced and steps 3 through 8 are repeated. When all submarines have been moved (or detected or scored) a check is made in decision box "all subs scored or detected" to determine if additional play is required.

10. Decision box "time" advances the time index until the game time selected has expired.

11. After each game time increment all sensors are moved according to their specified search plans. Then the positions of all sensors and submarines are stored before the next game move computations begin.

12. Steps 3 through 11 are repeated until game time expires or all submarines have scored or been detected. Before stopping, the Red and Blue score is entered in displayed index counters.

3.3 INPUT OF DATA:

Inputs may be of three forms, selected, entered, or programmed. "Programmed inputs" are those which involve changes in the data flow scheme; since these inputs require detailed consideration of the assembly routine language they will not be considered further in this section.

Normally, "entered data" consists of cutting paper "tapes" to enter position data, desired detection tables or search patterns, and similar inputs. Changes in entered data are encountered when changing major game features, or conditions. "Selected data" consists of selecting from among various entered or programmed tables, search patterns and parameters, or specifying simple parameter values such as game time. These inputs are normally made by entering in index registers or by the 1604 "Inspect and change routine". Changes in selected data are usually made for re-play of a specific game under various conditions, or to test parameter sensitivity.

See Appendix E for a detailed table of inputs.

3.4 OUTPUT OF DATA:

After completion of each game, the scores of Red and Blue are displayed in index registers. Additional information may be obtained by use of "inspect and change" or "print-dump". A résumé of the activity during each time interval and records of the sensor detections and submarine detections for play by play analysis are available for extraction.

See Appendix E for a detailed table of outputs.

3.5 DATA LIMITATIONS:

Because of the characteristics of the CDC 1604 computer there exists certain limitations such as the game time limit, number of sensors, etc. In general these limitations are not restrictive, and should cause no problem since real world situations do not approach these limitations. For instance, the maximum number of sensors allowed in this particular game is 48. These limitations are the result of word length in the computer. With appropriate program changes many of the existing limitations can be increased.

IV

SAMPLE APPLICATIONS

4.1 A WAR GAME:

To verify the gaming aspects, persons were chosen to act as Red and Blue Commanders and given the information listed under RULES. Each independently positioned his forces in a disposition which he thought optimal under existing conditions. The game was run and the results obtained are listed below. In addition an extraction of track data was made and plotted as shown in Figure 4-1.

Rules:

a) Blue - 6 destroyers, 5 helicopters

b) Red - 4 submarines

c) Game time limit - 90 minutes

d) Assured sonar range - 1500 yards

e) Classification probability - 50%

f) Score area and shoreline - as plotted in Figure 4-1

g) Temperature gradient - y coordinate 150

Results:

a) Actual computation time - 9 seconds

b) Blue scores - 2

c) Red scores - 2

d)

Submarine number	Detected by sensor number[1]	Time	Range
1	11	40	2100 yds.
2	not detected		
3	not detected		
4	4	06	2865 yds.

[1]Destroyers are numbered #1 through #6, and helicopters #7 through #11

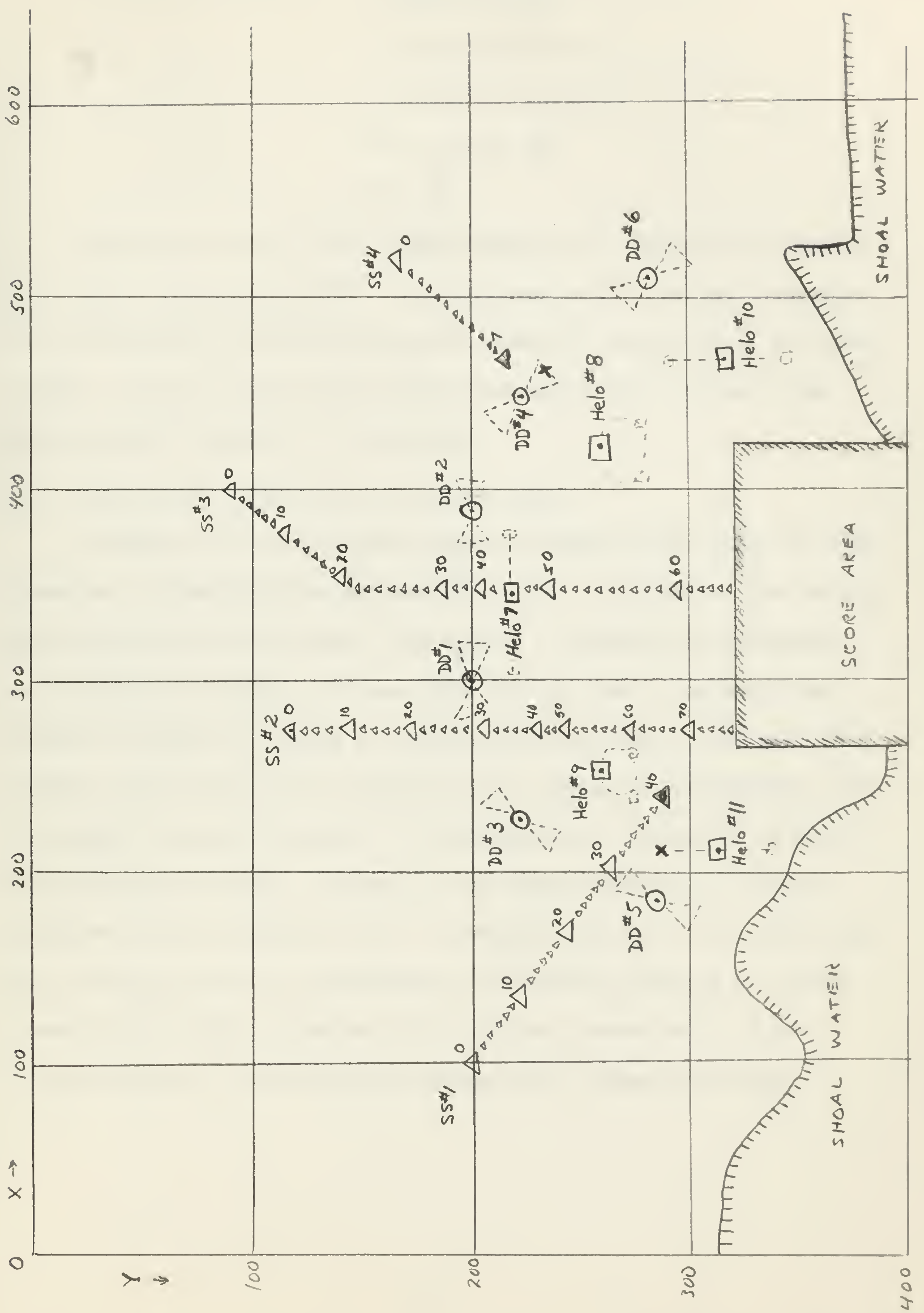


Figure 4-1. Sample Game

e) Submarine number	Sensors detected by submarine during its run
1	#3, #5, #7, #9, #11
2	#1, #3, #7, #9, #11
3	#1, #2, #7, #8
4	#2, #4

In Figure 4-1, the small numbers along the submarine track indicate the various times (in octal). The crossover patrol for the destroyers and the patrols of the helicopters are plotted, but times are not indicated. The "X" on the search pattern indicates the position of the sensor when it detected the submarine.

4.2 STATISTICAL VERIFICATION OF THE WAR GAME:

A sample of 20 runs was made using an assured sonar range of 1500 yards and a classification probability of .5. In Table 4-1 are listed the scoring results for runs 1 through 20. To verify the randomness of detection additional data was extracted for runs 1 through 10 and listed in Table 4-2. Table 4-2 showed that not only did the detecting sensor vary but also time of detection and submarine intelligence about the screen. Figure 4-2 gives the distribution of the number of detections for the 20 runs. The mean of the number detected is 2 and the estimate of the variance is .63. An analysis of data of the first ten runs indicates that the probability of detecting submarine #1 is 70%, submarine #2 is 50%, submarine #3 is 50%, and submarine #4 is 30%. In the 20 runs, 24 detections were made and 16 submarines scored.

TABLE 4-1

Run	SS Scored	SS Detected
1	2	2
2	3	1
3	2	2
4	0	4
5	4	0
6	2	2
7	2	2
8	1	3
9	2	2
10	1	3
11	1	3
12	2	2
13	2	2
14	3	1
15	2	2
16	2	2
17	1	3
18	3	1
19	2	2
20	3	1

TABLE 4-2

Sensor Detection Data

Run No.	SS#1 Sensor	SS#1 Time	SS#2 Sensor	SS#2 Time	SS#3 Sensor	SS#3 Time	SS#4 Sensor	SS#4 Time	Sensors Detected by Submarine SS#1	SS#2	SS#3	SS#4
1	11	40	-	-	-	-	4	06	3,5, 7,9, 11	1,3, 7,9, 11	1,2 7,8	2,4
2	-	-	-	-	2	26	-	-	3,5, 7,9, 11	1,3, 7,9, 11	2	2,4.8
3	3	33	3	50	-	-	-	-	3,5, 9,11	1,3, 7,8	1,2, 7,8	2,4,7, 8,10
4	11	26	1	14	2	34	4	07	3,5,9	1	2,7	2,4,10
5	-	-	-	-	-	-	-	-	3,5, 7.9, 11	1,3, 7,9, 11	1,2, 7,8	2,4,7. 8,10
6	-	-	1	53	2	45	-	-	3,5, 7,9, 11	1,3, 7,9	1,2, 7	2,4,7, 8,10
7	5	26	-	-	-	-	4	03	3,5	1,3, 7,9, 11	1,2, 7,8	---
8	3	44	-	-	8	65	4	03	3,5, 7,9, 11	1,3, 7,9, 11	1,2, 7,8	---
9	5	17	3	36	-	-	-	-	3	1,3, 7,9	1,2, 7,8	2,4,7, 8,10
10	3	35	7	37	7	26	-	-	3,5, 9	1,3, 7,9	2	2,4,9

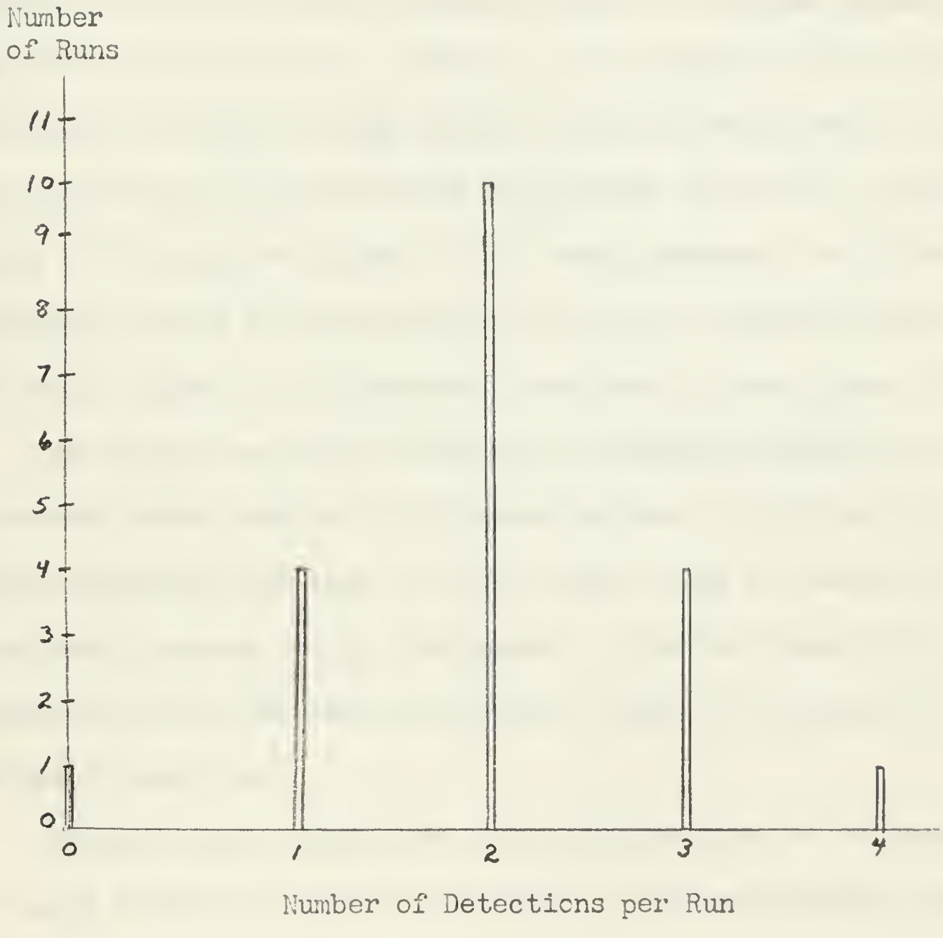


Figure 4-2. Distribution of Number of Submarines Detected per Run from a Sample of 20 Runs (4 submarines in each run)

4.3 DETECTION PARAMETER ANALYSIS:

An analysis was made to determine if using Sweep Width yielded the same results as using the actual curve of detection probability versus range.[1] The same inputs were used as in the War Game discussed in the previous sections.

Using a sample of twenty runs, no submarine was detected. An investigation of the track data indicated that no submarine passed within the Sweep Width of the sensors. That is, the submarine evasion routine functioned to keep the range greater than the Sweep Width. This indicated that when using the actual range curve detection is achieved primarily at the extreme limits of the search pattern. The detection advantage enjoyed by the submarine used in the evasion decisions accounts for this. Figure 4-3 illustrates the effect of using Sweep Width.

This theory was partly verified by performing additional runs using an assured sonar range of 3,000 yards in lieu of 1,500 yards, all other factors remaining the same. At this sonar range the Sweep Width circles of adjacent sensors nearly overlapped. Using both Sweep Width and the detection curve under these conditions resulted in detection of all submarines in each case.

Although these results are not exhaustive, there is evidence that for cases in which early evasive action may be encountered, reduction of search range curves to Sweep Widths is not justified.

4.4 TACTICAL DISPOSITION ANALYSIS:

To evaluate the effectiveness of alternate methods of employing the assigned helicopters the disposition of Blue forces was changed by

[1] The range versus detection probability function is integrated, and a range is determined within which the detection probability may be considered one; this range is the Sweep Width.

[illegible]

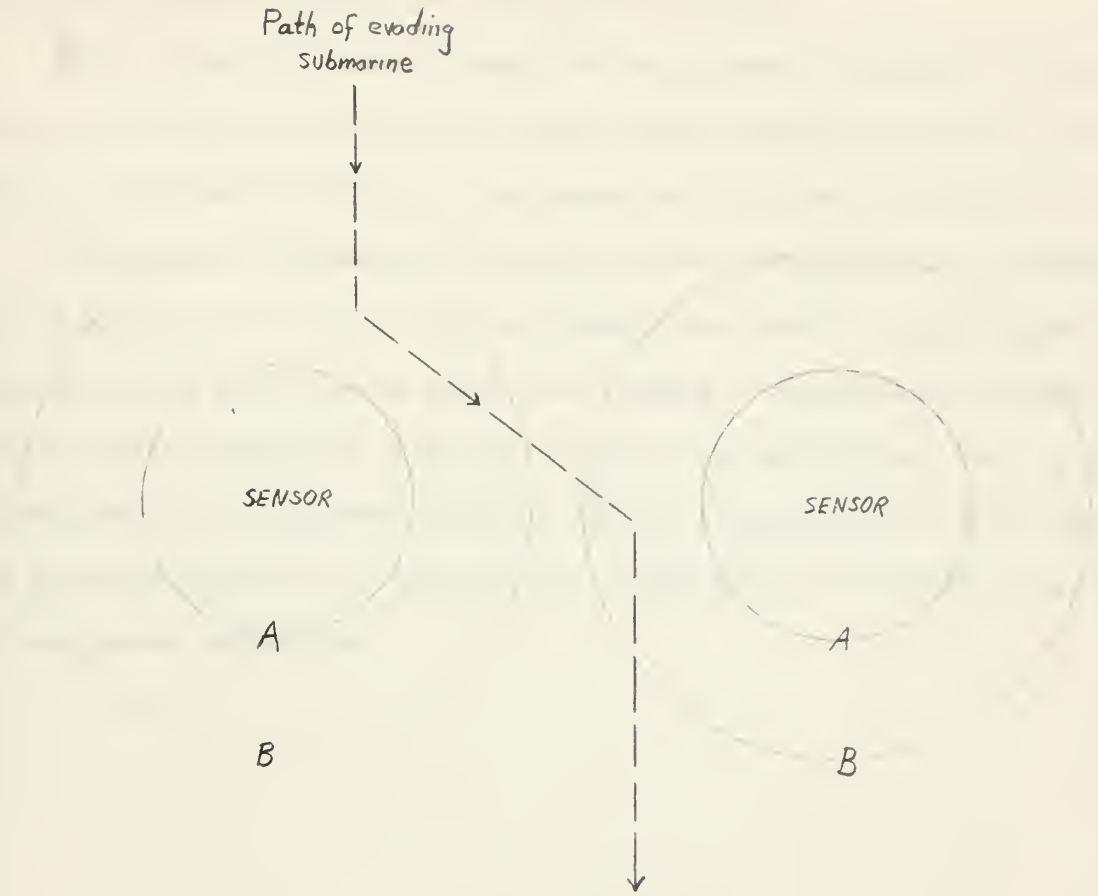
[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



A: Sweep Width, circles within which a detection probability of one exists.

B: Circles within which some probability (less than one) exists as a function of range.

Figure 4-3. Effect of using Sweep Width

stationing the helicopters outside the destroyer barrier. All other inputs were again the same as used in the War Game.

Using a sample of twenty runs, the mean number of submarine detections achieved was 1.85 per run, with a sample variance of 1.55. This data is presented in Table 4-3 for comparison with that of Table 4-1.

Statistical comparison of the two tactical dispositions is amenable to the use of a Chi-Square Test for Enumeration Data.(6) Applying this test at the 5% significance level, the numbers of detections achieved using the new tactical disposition, in which the helicopters are deployed in advance of the destroyer barrier, did not significantly differ from the previous disposition (Figure 4-1). Hence the new disposition can not be considered preferable.

TABLE 4-3

Run	SS Scored	SS Detected
1	1	3
2	1	3
3	0	4
4	2	2
5	2	2
6	1	3
7	2	2
8	2	2
9	1	3
10	3	1
11	3	1
12	1	3
13	2	2
14	3	1
15	4	0
16	1	3
17	3	1
18	2	2
19	4	0
20	4	0

V

CONCLUSIONS

5.1 CRITIQUE:

Although major emphasis has been placed on the investigation of gross parameters, the gaming aspect also plays a prominent part. Not only is there great training value for defensive planning but also valuable experience for the offensive player. With given initial inputs each player positions his forces in an optimal configuration in order to get maximum payoff. The computer resolves this conflict of interest and determines the payoff for each. Of greater interest is the distribution of the scores rather than the results of a single run. Although an analysis of 20 runs with a single set of parameters gave results as shown in Figure 4-2, outcomes are entirely dependent upon the initial inputs.

The ultimate effectiveness for this game will be its conformity with a real world situation. The original purposes for which the game was constructed have been accomplished. Its speed and flexibility of control are proven factors. The ease with which given conditions may be duplicated and repeated trials conducted with slight modifications demonstrate that this method is feasible and relatively inexpensive. As previously mentioned its ultimate effectiveness will be realized when the data from an actual operation may be run and a comparison of actual and computer results made.

It should be reemphasized here that the ASW Sortie problem presented represents only a vehicle for displaying the particular methodology proposed by this thesis. These methods may equally well be applied to an air defense problem, or a strategic battle problem. While it is hoped that the game as programmed may have some value (with and without

modifications), our prime concern in this thesis has been to emphasize broad principles at the expense of the details of programming and operating the game.

The complete CDC 1604 assembly routine program, detailed flow charts and operating instructions are available upon request to the United States Naval Postgraduate School, Monterey, California, Attention Professor R. M. Thatcher.

5.2 METHODS OF IMPROVEMENT

As was originally stated this is a simplified version of a real world war game. It has value in its present form to investigate gross parameters and to be used as a training device. There are many ways in which this game may be sophisticated to conform more closely to a real life situation.

(a) Modify the search pattern to conform to the destroyer pattern and the helicopter pattern so that it can detect only in one quadrant each game minute. This will approximate scanning sonar.

(b) All units employed in the game should not have the same capability of detecting and classifying. A state of training correction is needed. This correction should be in the interval (0,1) and choice for each unit could be made by some random method. The correction would then be used to modify the probabilities of detecting and classification.

(c) As no screen is completely synchronized, an out of phase relation should be introduced for each sensor. This could be accomplished by comparing a generated random number against an interval. If the random number is less subtract one minute

from the destroyer movement time, if in the interval, no correction, and if greater than the interval, add one minute to the destroyer time.

(d) Detection should not be based entirely on the closest sensor but must use all sensors that are capable of detecting. This can be accomplished as in radar problems by finding the product of the probability of not detecting and subtracting from one. The same would apply to classification.

(e) Considerable time is required to plot and analyze the track data from the 5000 or more cells in the "dump". A map printing similar to the NANWEP routine would eliminate this tedious aspect. Or by an alternate method, since the online printer will print four cells to a line, a 32 mile grid would require each cell to represent 8 miles in length. Each hexagonal grid is 160 yards and with the 16 digit word each octal digit would represent $\frac{1}{2}$ mile. Insertion of a number from 1 to 6 would further break each of these $\frac{1}{2}$ mile digits into an area of approximately 166 yards in length. The above procedure plus the usual "dump" would provide all the information needed for analysis.

(f) More flexibility of sensor search patterns is needed. Provisions should be made to move helicopter positions during the run. For example the helicopters could be placed outside the destroyer screen for the first 45 minutes of the game and then automatically moved inside. This could be accomplished by comparing the time index against a certain constant. When the time index exceeds the constant enter a new set of helicopter positions in the sensor permanent positions. (Note sensor

temporary position is always found by addition of a movement table entry to the permanent position.)

(g) This game was initially conceived as a game of detection only. It has a limitation which could be rectified. A sensor making contact would normally depart his station to investigate and thereby create a temporary hole in the barrier. A procedure should be included after the positive classification to remove the sensor from his position, stop the game and notify the Blue Commander which sensor has departed his station. Blue could plug the hole by a time delay method. The game could continue for any specified time at which another stop could occur to notify Blue that his sensor is ready to return to his old position.

VI

RECOMMENDATIONS

Ths increasing emphasis on machine war games and the current trend toward a universal computer language indicate that the utility of this game could be greatly improved by the following:

1. Expand and improve the game to eliminate some of the simplifications and assumptions.

2. Reprogram the game in FORTRAN[4] or NELIAC[5] compiler language.

3. Investigate more fully the effect of submarine evasive action upon approximation of the detection problem by using Sweep Width instead of the actual detection probability curve.

BIBLIOGRAPHY

1. R. E. Zimmerman, A Monte Carlo Model for Military Analysis, Operations Research for Management, Volume II, John Hopkins Press, Baltimore, Maryland, 1956.

2. OPNAV Instruction 03000.7 U. S. Navy War Game Program.

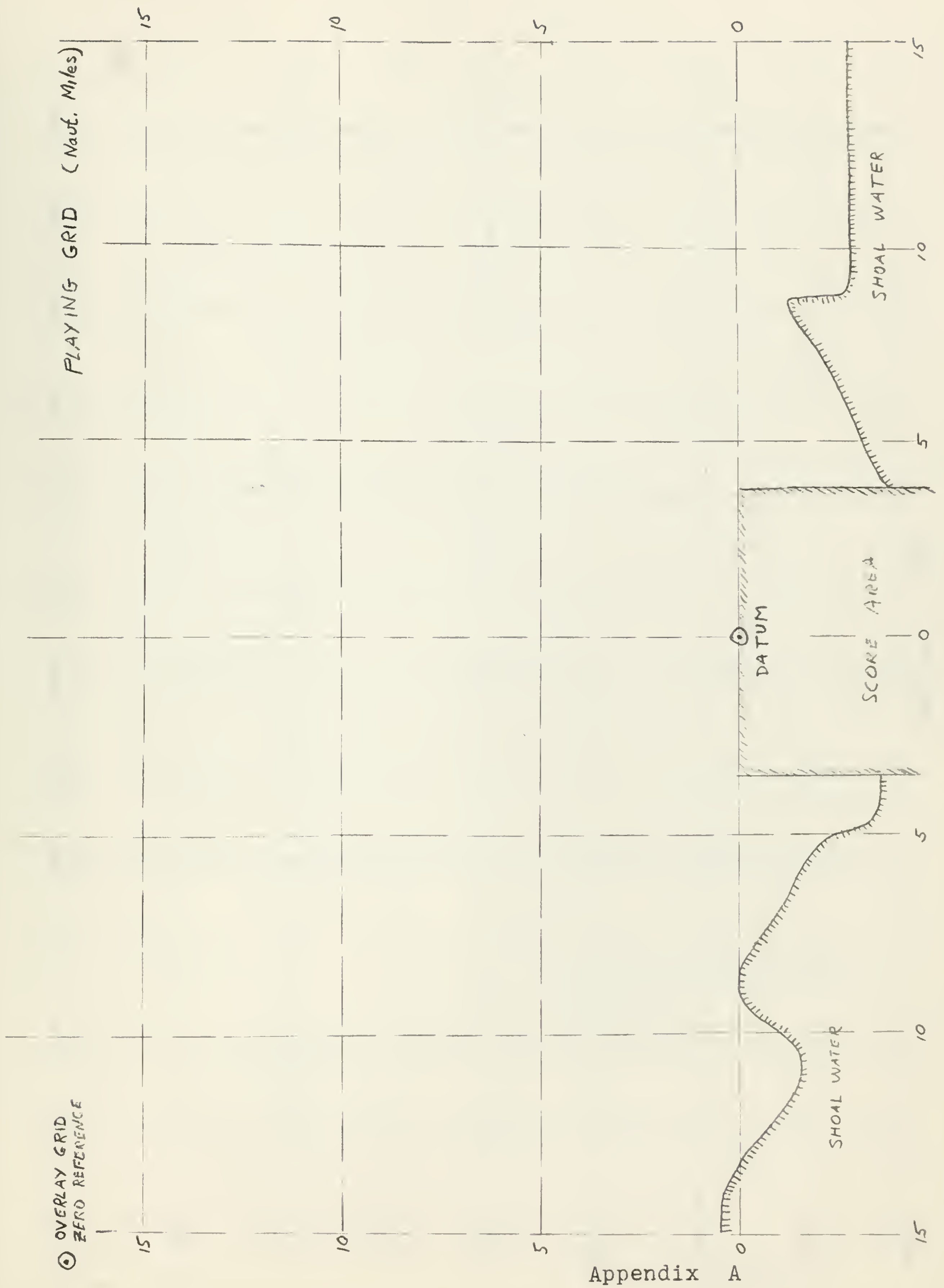
3. Control Data Corporation, Characteristics of the Model 1604 Computer, Publication 018b.

4. Control Data Corporation, FORTRAN SYSTEM, Publication 087A.

5. R. M. Thatcher, The NELIAC Compiler Language, CDC-1604 Version, dated September 1960, U. S. Naval Postgraduate School, Monterey, California.

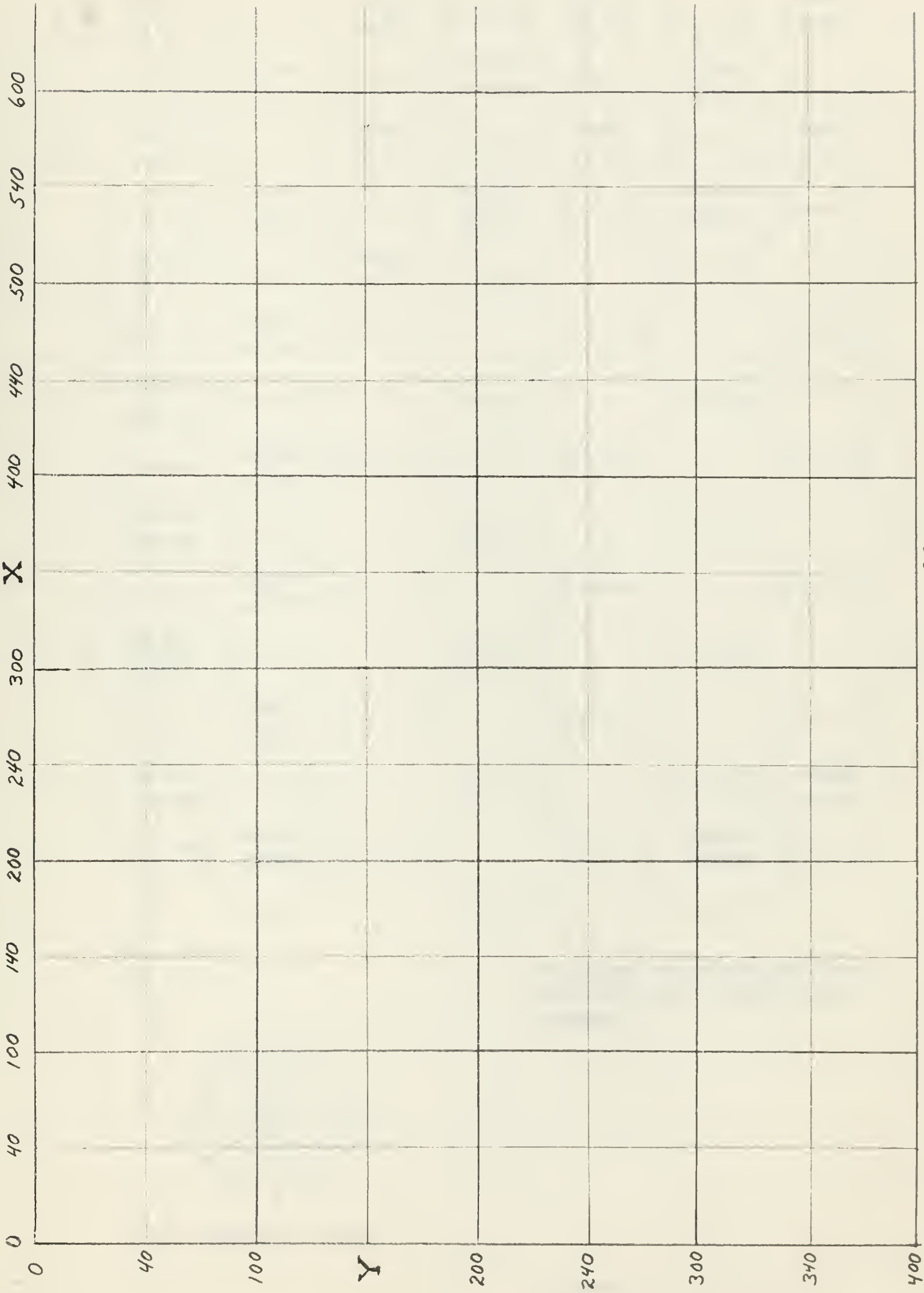
6. A. H. Bowker and G. J. Liberman, Engineering Statistics, Prentice-Hall, Englewood Clifts, N. J., 1960.

7. R. D. Specht, War Games, Rand Corporation, P-1041, 1957.


PLAYING GRID (Naut. Miles)
OVERLAY GRID ZERO REFERENCE
DATUM
SCORE AREA
SHOAL WATER
SHOAL WATER
15
10
5
0
5
10
15

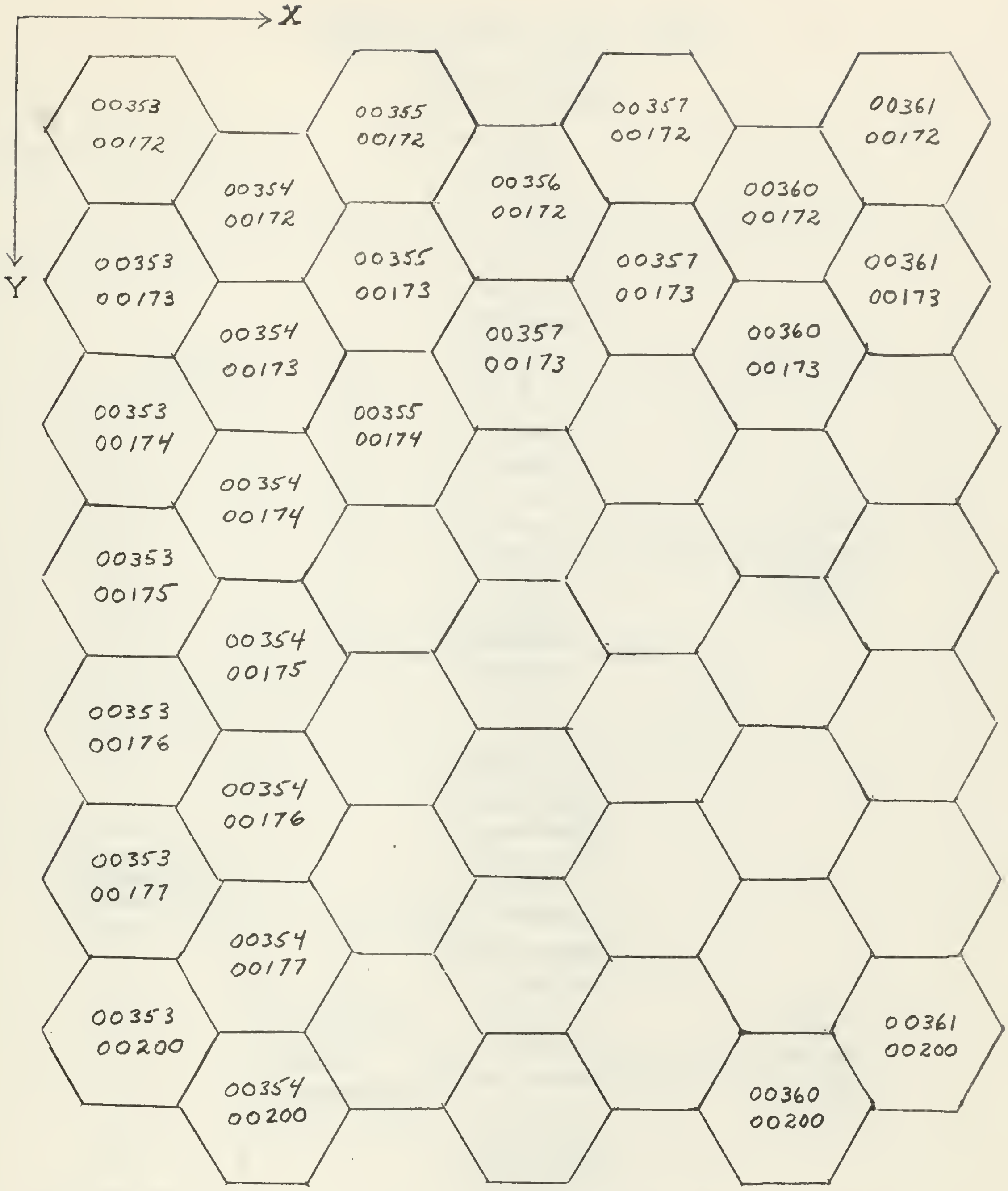
Appendix A

OVERLAY GRID

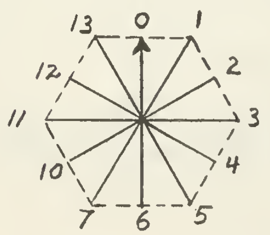


(OCTAL CELLS)

Appendix B



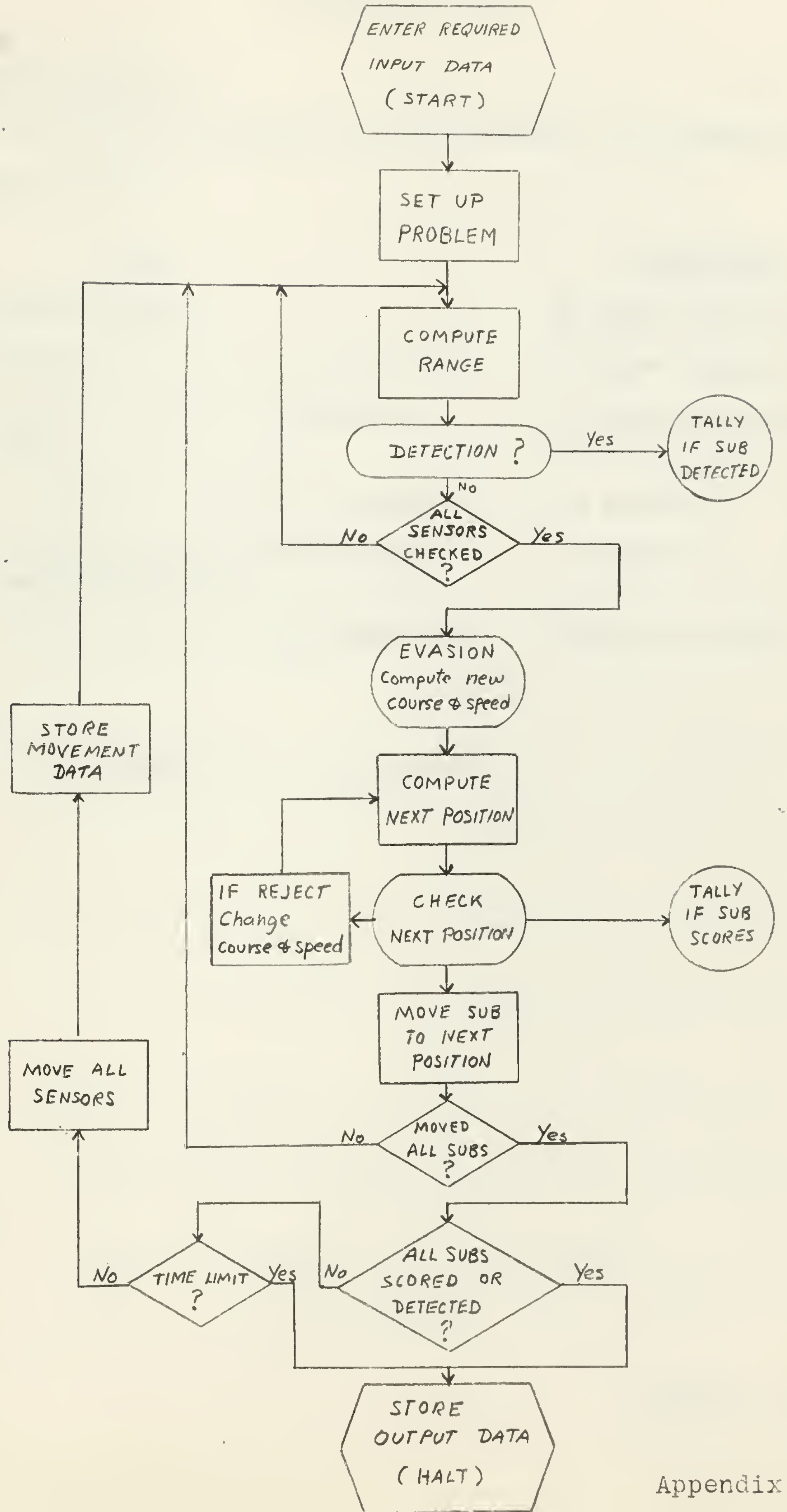
Portion of Overlay Grid showing grid numbering scheme



Grid Compass Rose

Appendix C

BLOCK DIAGRAM FLOW CHART



Appendix D

INPUT-OUTPUT TABLE I

Selected Input Data

The following data may be selected readily by the operator at the computer console:

Data	Form of Data
(1) Game time limit	one index register entry
(2) Choice of entered sonar range curve	one index register entry
(3) Choice of entered classification curve	one index register entry
(4) If elimination of Résumé is desired	one jump key
(5) If storage of all generated random numbers is desired	one jump key
(6) Address to start storing Résumé data	accumulator entry

Note:

(4) See Detailed Output Data section.

[illegible]

INPUT-OUTPUT TABLE II

Entered Input Data

The following data may be entered into the program by paper tape or by the CDC 1604 "Inspect and Change Routine":

Data	Form of Data
(1) Grid dimensions	two memory cell values
(2) Score area dimensions	four memory cell values
(3) Stress time	one memory cell value
(4) Detection anomaly parameter	one memory cell value
(5) Initial random number	two memory cell values
(6) Submarine speed cut-off for evasion	one memory cell value
(7) Starting position cells of submarines	number of subs plus two memory cell values
(8) Reference position cells of sensors	number of sensors plus two memory cell values
(9) Shoal water line	table of Y values (one for each X in overlay grid)
(10) Sonar detection range curves	five tables (total) each of 63 cells
(11) Classification curves	three tables (total) each of 63 cells
(12) Sensor search pattern tables	eight tables (total) each of 26 cells

Notes:

(1)(9) The shoal water line must be consistent with the grid dimensions (but see next note)

(2)(9) If used as an ocean barrier (i.e. without shoreline) score area X dimension extends over the entire grid; in this case a shoal water line table is not required.

(4) To make the Y dimension of the detection anomaly parameter a variable, or to reverse the anomaly effect in order to simulate shoreline reverberation requires program changes.

(7)(8) This data must be of the form shown in Appendix G.

INPUT-OUTPUT TABLE III

Displayed Output Data

After the computer has played one game the following data is displayed visually on the console:

Data	Form of Data
(1) Number of submarines having scored	one index register
(2) Number of submarines having been detected	one index register
(3) Last random number generated	accumulator

Notes:

(1)(2) Although the game is programmed for one play, with a minor program addition multiple plays for the same input data may be achieved. The scores would then be of a statistical nature rather than a tally.

(1)(2) It is possible that a sub may neither score nor be detected.

(3) To avoid single bit numbers and other vagaries which might void the random number generation.

INPUT-OUTPUT TABLE IV

Detailed Output Data

If selected during input, the following data is available for observation by "Inspect and Change Routine", or for flexowriter or line-printer dump:

Data	Form of Data
(1) Resume´	Stored consecutively starting at selected address— a. Game time. b. All sensor position cells (at that game time) in numerical order. c. All submarine position cells in numerical order.
(2) All random numbers generated	Stored consecutively from fixed address.
(3) Submarine detection record	One cell for each sub (in numerical order) showing the detecting sensor number and the game time of detection.
(4) Submarine evasion record	One cell for each sub showing sensors the sub detected during penetration.

Notes:

(1) Position cells are of the form of Appendix G.

(2) Considerable memory space may be required for this storage so care must be exercised not to ruin stored data or parts of the program. It is anticipated that this feature will find application only during special analytical tests.

(4) The octal cell value obtained must be changed to a binary number. A one bit indicates detection. The sensor number is obtained by counting from the right edge.

[illegible]

[illegible]

[illegible]

[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]

[illegible]

[illegible]

[illegible]

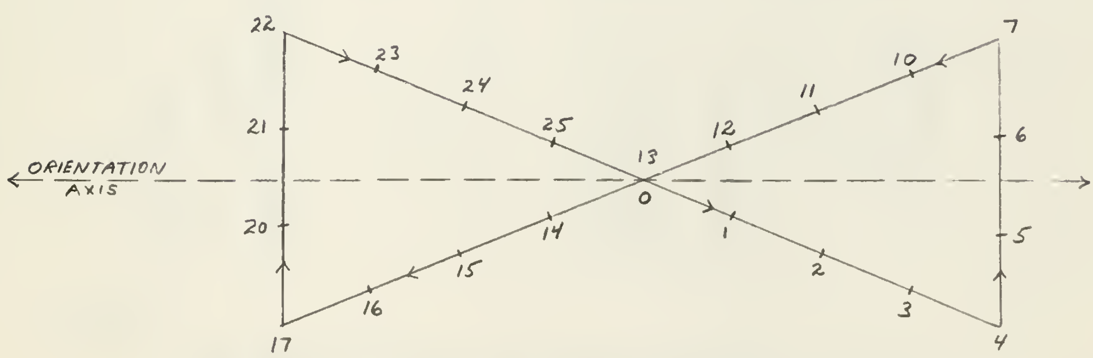
[illegible]

DESTROYER SEARCH METHOD

Portion of destroyer movement table:

t_c	$\Delta\dot{x}$	$\Delta\dot{y}$	C_D
0	0	0	4
1	+3	+1	4
2	+6	+2	4
3	+11	+3	4
4	+14	+1	0
5	+14	−1	0
6	+14	−3	0
7	+14	−3	10
10	+11	−1	10
|	|	|	|
|	|	|	|
|	|	|	|

Typical Search Pattern:



Note: At reference position t_c equals zero and 13.

SAMPLE OF POSITION CELL PACKING

Submarine:

0	0	4	0	0	3	4	2	3	0	5	0	0	1	7	6
Always Zero	C_i (0 to 13)		X_s (0 to 37777)					V_s (0 to 5)	C_z (0 to 13)		Y_s (0 to 37777)				

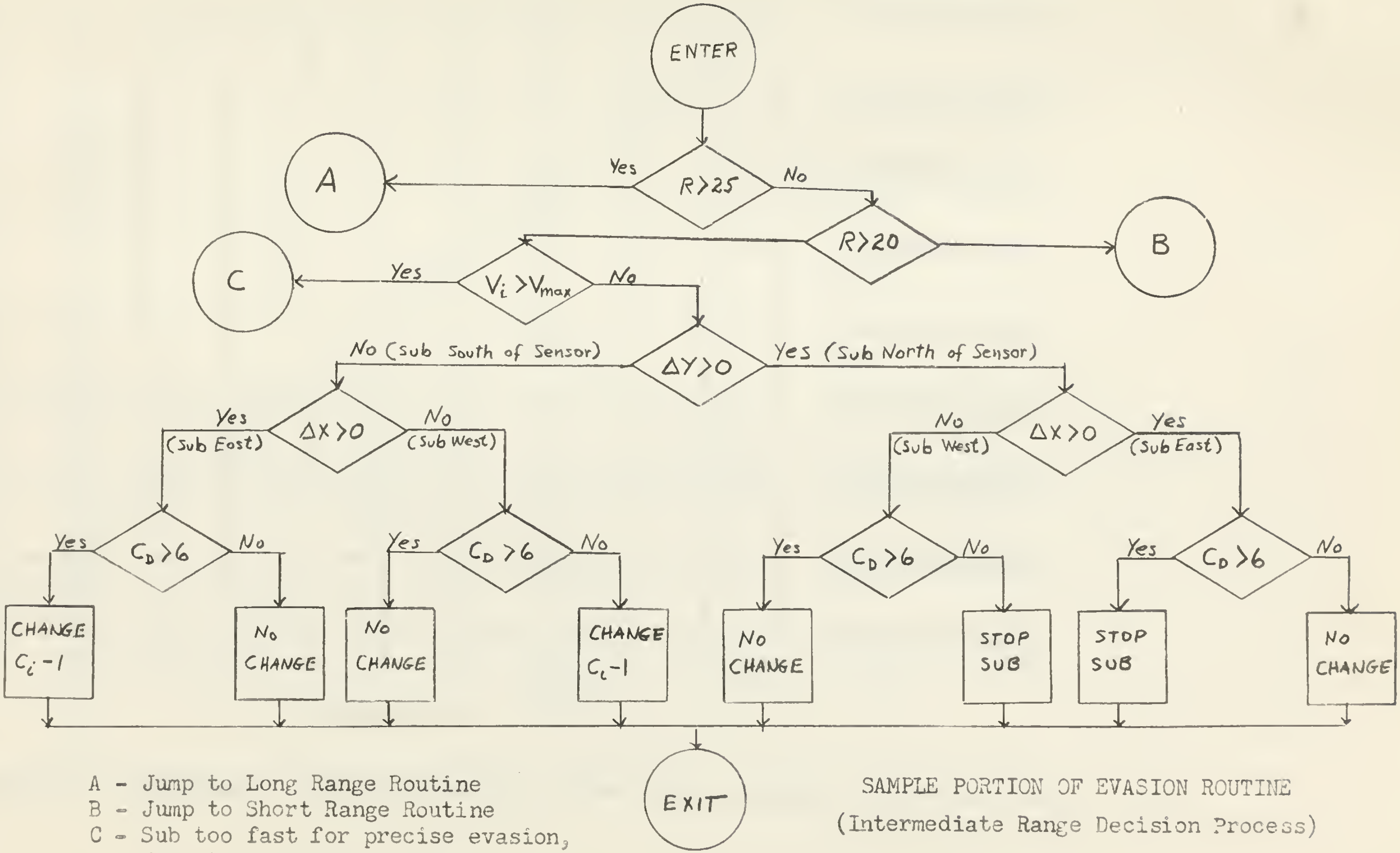
Destroyer:

0	1	0	0	0	1	2	4	1	0	0	0	0	2	3	4
Always Zero	C_D (0 to 13)		X_D (0 to 37777)					σ_D (0 to 4)	Always Zero		Y_D (0 to 37777)				

Helicopter:

0	7	7	0	0	5	3	6	6	0	0	0	0	0	6	2
Always Zero	77 is code to identify as helicopter		X_H (0 to 37777)					σ_H (5 to 7)	Always Zero		Y_H (0 to 37777)				

Notes:
(1) Entries are in octal digits
(2) All input and output data is of this form



A - Jump to Long Range Routine
B - Jump to Short Range Routine
C - Sub too fast for precise evasion,
    Jump to Long Range Routine

SAMPLE PORTION OF EVASION ROUTINE
(Intermediate Range Decision Process)

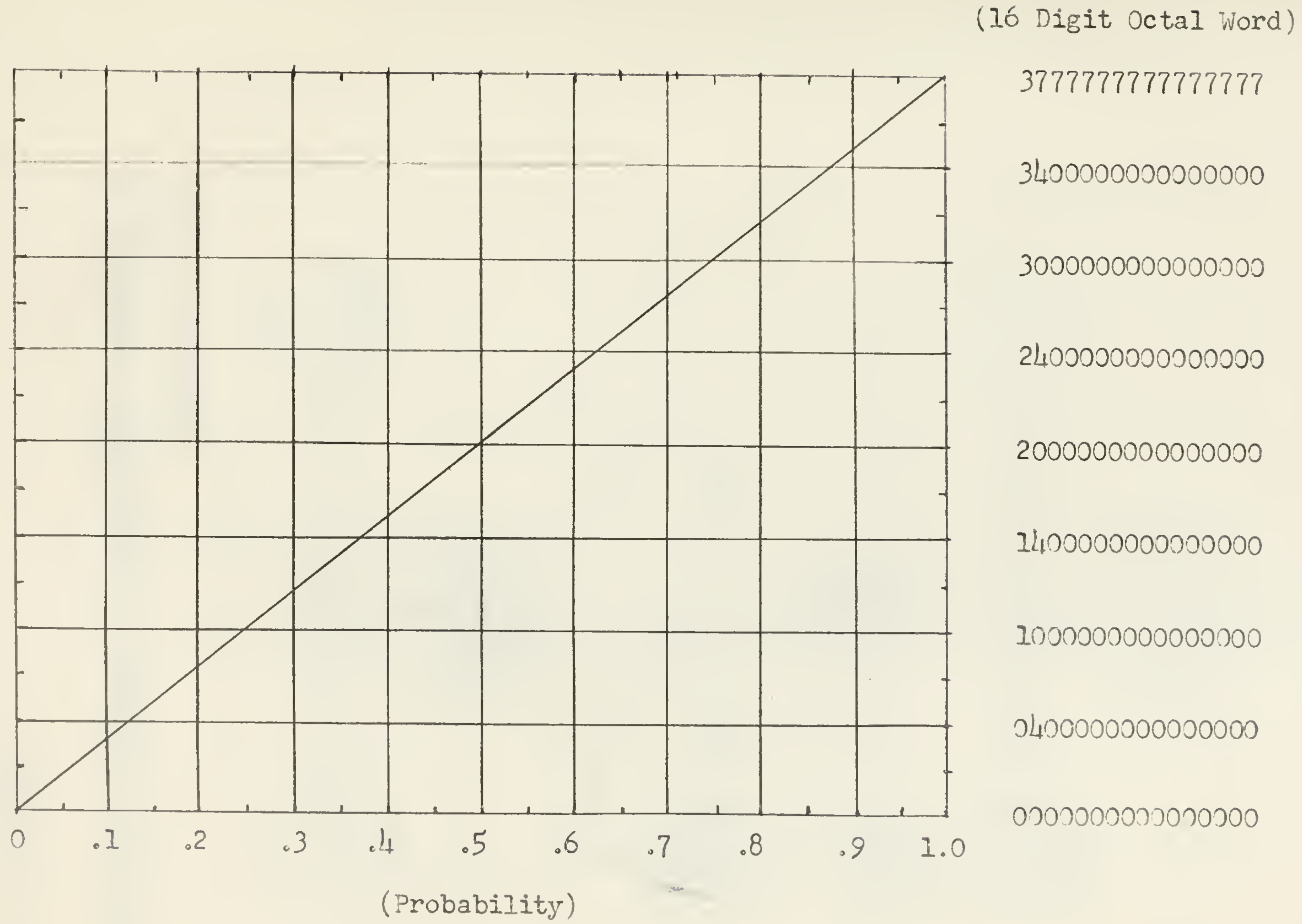
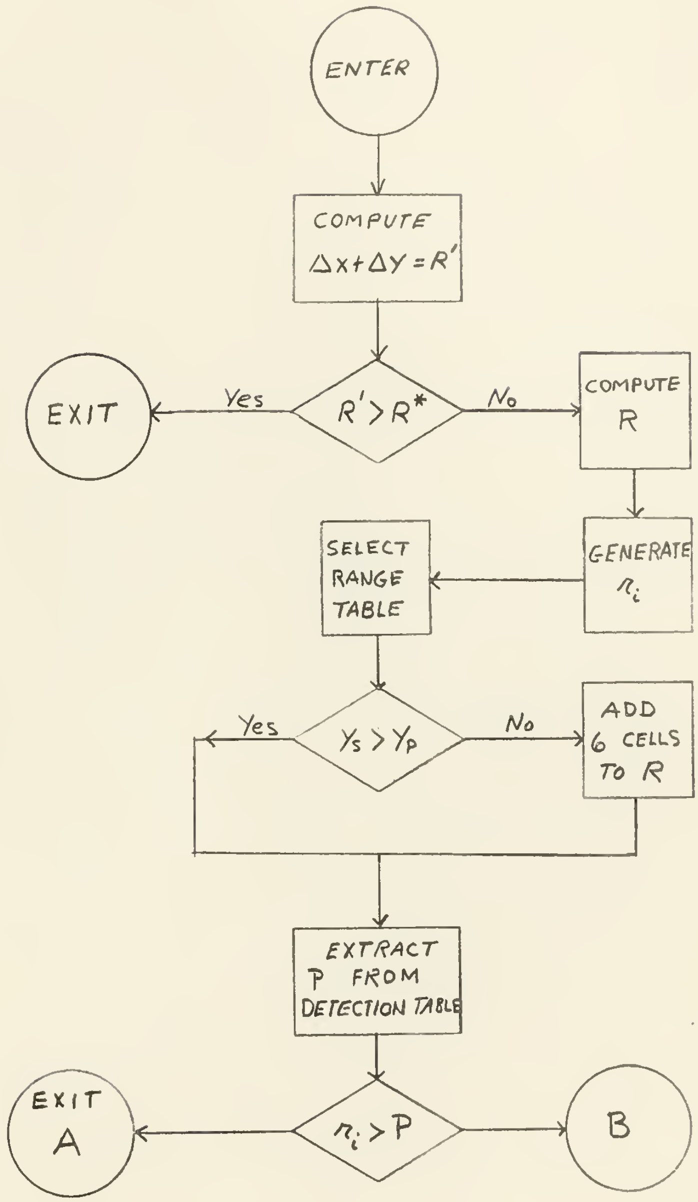


Table for Conversion of Desired Probability Value into a CDC 1604 Octal Table Entry

Appendix I

SAMPLE PORTION OF DETECTION ROUTINE



A- No detection

B- Continue computation, submarine detected